CS4133/5133: Data Networks

Fall 2017

100 pts

Project-1

Client-Server System

Due Date: Oct 01, 2017

1. Objective

The objective of this project is to learn TCP client-server interaction using socket interface in C programming language. After completing this Project, you will have a basic understanding of the steps required to develop a networking application.

2. Problem Specification

In this project, you are required to do socket programming in C language (Linux environment) to implement a pair of client and server that can achieve simple password verification to indicate the current market value of the stock. Your client will send a pair of username and password to your server and your server will verify whether the pair of username and password is legitimate or not for retrieving the market value. Assume the only legitimate pairs of usernames and passwords that will be accepted by your server are as follows.

Username	Password	Value
apple	jP79bNs2	158.63
cisco	m6W7p8uv	31.48
facebook	J46H6T1b	170.95
google	nCh781fR	941.41
intel	ad82bG54	35.19
microsoft	tfw61RqJ	73.98

Specifically, your client and server programs entail to achieve the following requirements:

1. Your client program needs to take two arguments that specify the name of server and the port that it is trying to connect to. Your program for server needs to take an argument that specifies the port that it is listening to. You can use (*5000+last 4 digits* of your student-id number) to avoid requesting same port by multiple students.
2. The server program will start first and keep listening to the specified port. Your client will connect to the port that your server is listening to, and a socket between your client and server is constructed.
3. Your client program will first prompt a welcome message that asks the user to enter a username using the keyboard. This username will then be sent to the server. Then, your

server, after receiving the username from your client, will send an acknowledgment message to the client.

4. Your client, after receiving the acknowledgment message from your server, will prompt a message that asks the user to enter the corresponding password. This password will then be sent to the server. Then, your server, after receiving the password from your client, will verify the received pair of username and password against the list of legitimate pairs. If the result is positive, the server will send a success message along with the current market value of the stock to the client. If the result is negative, the server will send a failure message to the client.
5. Your client, after receiving the result message, will print out the result and close the socket. Your server will close the socket following the client, and keep listening for the next client request.
6. Your server will close the socket after waiting for the username or the password for 30 seconds.

3. Programming Notes:

I suggest you to start modifying the TCP server program given in Canvas first. Your server program needs to handle FILE I/O for authentication. When the server is up and running, you can start modifying the TCP client program also given in Canvas.

This project will help you to understand the basic client-server interaction using TCP sockets. You can also enhance the program by converting your server a concurrent server to handle multiple clients at the same time. Some portions of the project1 might be needed for upcoming projects.

File names:

Make sure you follow the file name guideline given below for your project:

lastNamep1c.c, and lastNamep1s.c

For example, if your Last Name is Boren, then file name should be: borenp1c.c for client program and borenp1s.c for server program.

4. Points Distribution:

Bits and pieces	Points
Client Program	40
Server Program	40
Program Style (Coding style, Comments etc.)	10
Documentation	10

5. Submission Instructions:

This project requires the submission of a *soft copy* and a *hard copy.*

5.1. Soft Copy (Due October 01, 2017, 11:59 pm)

The soft copy should consist of the server, client and any other header file(s) along with the detailed documentation of the programs. Documentation should include your problem-solving approach, discussion of data structures, algorithms, user define functions, and screen shots of outputs. These must be submitted through Canvas (http://canvas.ou.edu). You have to make sure that your program works with the file format given in this project description. If I need any clarification regarding your coding, you will need to come at my office hours to demonstrate and/or run the code.

5.2. Hard copy (Due October 02, 2017, beginning of the class)

The hard copy of the project, consisting of the items submitted as soft copy, should be submitted at the beginning of the class. The hard copy must be the same as the soft copy.

Please **DO NOT MODIFY** the project code in any way after the deadline of soft copy submission.

6. Late Penalty:

You have to submit your project on or before the due date to avoid any late penalty. **A late penalty of *15% per day* will be imposed after the *due date (Oct 01, 2017)***. After one week from the due date, you will not be allowed to submit the project.

Good Luck!!